FOCUS Curvature Objective Functions

Thomas Kruger

April 21, 2020

General

FOCUS has three curvature objective functions implemented. The linear objective function, $f_{curv,1}$ is given in (1) and the quadratic objective function, $f_{curv,2}$ is given in (2). The third objective function, $f_{curv,3}$ given in (3) uses a penalty method to apply a constraint on the minimum radius of curvature, $min\ R_c = \frac{1}{\kappa_0}$ where κ_0 is a user-specified maximum curvature. For this to be a true penalty function and constraint on curvature, α would need to be iterativly increased to ∞. This objective function works surprisingly well with $\alpha = 2$ and an appropriate objective function weighting that iteratively increasing α is not implemented in FOCUS. Typically the constraint is either not violated or only violated by a small fraction of κ_0.

$$f_{curv,1} = \frac{1}{N_c} \sum_{i=1}^{N_c} \int_0^{2\pi} \kappa_i \, dt \tag{1}$$

$$f_{curv,2} = \frac{1}{N_c} \sum_{i=1}^{N_c} \int_0^{2\pi} \kappa_i^2 \, dt \tag{2}$$

$$f_{curv,3} = \frac{1}{N_c} \sum_{i=1}^{N_c} \int_0^{2\pi} H_{\kappa_0}(\kappa_i) \left(\kappa_i - \kappa_0\right)^{\alpha} dt \tag{3}$$

$$\alpha \geq 2 \qquad \kappa_i,\, \kappa_0 > 0$$

$$H_{\kappa_0}(\kappa_i) \equiv H(\kappa_i - \kappa_0) = \begin{cases} 0, & \kappa_i < \kappa_0 \\ \frac{1}{2}, & \kappa_i = \kappa_0 \\ 1, & \kappa_i > \kappa_0 \end{cases} \tag{4}$$

First Derivatives

First derivatives of the three objective functions are now given. Functional derivatives are calculated down to derivatives with respect to parameterizing variable. The large number of terms in the functional derivative makes implementation difficult. Because of this, mixed partial derivatives of parameterizing variable and Fourier mode amplitude are given. If additional parameterizations are implemented and optimized, mixed partial derivatives will need to be calculated and implemented. Here X_i is an arbitrary variable, $X_i \in \mathbf{X_i} = \{X_{c,0}, ..., X_{c,NF}, X_{s,1}, ...X_{s,NF}\}$, that defines the x position of the ith coil. Derivatives for a Y_i or Z_i take a similar form.

$$\frac{\partial f_{curv,1}}{\partial X_i} = \frac{1}{N_c} \int_0^{2\pi} \frac{\partial \kappa_i}{\partial X_i} \, dt \tag{5}$$

$$\frac{\partial f_{curv,1}}{\partial X_i} = \frac{1}{N_c}\int_0^{2\pi} \frac{\delta K_i}{\delta x_i}\frac{\partial x_i}{\partial X_i}\, dt \tag{6}$$

$$\frac{\partial f_{curv,2}}{\partial X_i} = \frac{1}{N_c}\int_0^{2\pi} 2\kappa_i \frac{\partial \kappa_i}{\partial X_i}\, dt \tag{7}$$

$$\frac{\partial f_{curv,2}}{\partial X_i} = \frac{1}{N_c}\int_0^{2\pi} 2\kappa_i \frac{\delta K_i}{\delta x_i}\frac{\partial x_i}{\partial X_i}\, dt \tag{8}$$

$$\frac{\partial f_{curv,3}}{\partial X_i} = \frac{1}{N_c}\int_0^{2\pi} \alpha\, H_{\kappa_0}(\kappa_i)\,(\kappa_i - \kappa_0)^{\alpha-1}\, \frac{\partial \kappa_i}{\partial X_i}\, dt \tag{9}$$

$$\frac{\partial f_{curv,3}}{\partial X_i} = \frac{1}{N_c}\int_0^{2\pi} \alpha\, H_{\kappa_0}(\kappa_i)\,(\kappa_i - \kappa_0)^{\alpha-1}\, \frac{\delta K_i}{\delta x_i}\frac{\partial x_i}{\partial X_i}\, dt \tag{10}$$

i indices are dropped for convenience.

$$\frac{\delta K}{\delta x} = \frac{\partial \kappa}{\partial x} - \frac{d}{dt}\frac{\partial \kappa}{\partial x'} + \frac{d^2}{dt^2}\frac{\partial \kappa}{\partial x''} \tag{11}$$

$$\kappa = \frac{\left((z''y' - y''z')^2 + (x''z' - z''x')^2 + (y''x' - x''y')^2\right)^{1/2}}{\left(x'^{\,2} + y'^{\,2} + z'^{\,2}\right)^{3/2}} \tag{12}$$

$$\frac{\partial \kappa}{\partial x} = 0 \tag{13}$$

$$\begin{aligned}\frac{\partial \kappa}{\partial x'} = {} & \frac{\left((z''y' - y''z')^2 + (x''z' - z''x')^2 + (y''x' - x''y')^2\right)^{-1/2}\left(y''(y''x' - x''y') - z''(x''z' - z''x')\right)}{\left(x'^{\,2} + y'^{\,2} + z'^{\,2}\right)^{3/2}} - \\ & \frac{3x'\left((z''y' - y''z')^2 + (x''z' - z''x')^2 + (y''x' - x''y')^2\right)^{1/2}}{\left(x'^{\,2} + y'^{\,2} + z'^{\,2}\right)^{5/2}}\end{aligned} \tag{14}$$

$$\frac{\partial \kappa}{\partial x''} = \frac{\left((z''y' - y''z')^2 + (x''z' - z''x')^2 + (y''x' - x''y')^2\right)^{-1/2}\left(z'(x''z' - z''x') - y'(y''x' - x''y')\right)}{\left(x'^{\,2} + y'^{\,2} + z'^{\,2}\right)^{3/2}} \tag{15}$$

To use functional derivatives, first derivatives of (14) need to be calculated as well as second derivatives of (15). Due to the number of terms in these derivatives, mixed partial derivatives were chosen to be implemented in FOCUS.

$$\begin{aligned}\frac{\partial \kappa}{\partial X} = {} & \left((z''y' - y''z')^2 + (x''z' - z''x')^2 + (y''x' - x''y')^2\right)^{-1/2} \\ & \left((x''z' - z''x')\left(\frac{\partial x''}{\partial X}z' - z''\frac{\partial x'}{\partial X}\right) + (y''x' - x''y')\left(y''\frac{\partial x'}{\partial X} - \frac{\partial x''}{\partial X}y'\right)\right)\left(x'^{\,2} + y'^{\,2} + z'^{\,2}\right)^{-3/2} - \\ & 3\left((z''y' - y''z')^2 + (x''z' - z''x')^2 + (y''x' - x''y')^2\right)^{1/2}\left(x'^{\,2} + y'^{\,2} + z'^{\,2}\right)^{-5/2} x'\frac{\partial x'}{\partial X}\end{aligned} \tag{16}$$

$$\frac{\partial x'}{\partial X_{c,\tilde{n}}} = -\,\tilde{n}\sin(\tilde{n}t) \tag{17}$$

$$\frac{\partial x''}{\partial X_{c,\tilde{n}}} = -\,\tilde{n}^2\cos(\tilde{n}t) \tag{18}$$

Notes

Using curvature objective functions allows for more local details to be in the coils, while solving for coils that are more reasonable to engineer and manufacture. Because of this, a higher number of Fourier modes can be included in the optimization. Before curvature objective functions were implemented, NF around 8 solved for reasonable coils, where NF is the maximum Fourier mode amplitude. With curvature objective functions, NF can be increased to around $NF = 24$ or higher. At some point including more Fourier modes will not benefit the optimization. The key point is that NF is not bound by coil complexity if curvature objective functions are used.

Since curvature is a local value, it is important to discretize the coils appropriately. Converge checks on curvature should be performed to find an appropriate $Nseg$ value, where $Nseg$ is the number of segments in a coil. Before these curvature objective functions, a value of $Nseg = 128$ was appropriate. With curvature objective functions, a fully converged discretization was around $Nseg = 384$.

How to Use

To use any of the three curvature objective functions, a weight, *weight_curv*, needs to be set. This variable and all other variables in this section are set in the "*.input" file. No changes to the "*.focus" file are necessary. The variable, *case_curv*, determines which curvature objective function is used. Valid options for *case_curv* are $1, 2, 3$. If $case\_curv = 3$ then the third curvature objective function, (3) is used and two additional variables need to be set. The variable $k0$ needs to be set to some positive non integer value. For HSX a reasonable value for $k0$ is 12.0. The α value in (3) also needs to be set to some non integer value that is greater than or equal to two. A reasonable choice for α is 2.0. The variable name for α is *curv_alpha*.